SSP and Sound Description
Author(s): Paul Berg, Robert Rowe and David Theriault
Source: *Computer Music Journal*, Vol. 4, No. 1 (Spring, 1980), pp. 25-35
Published by: The MIT Press
Stable URL: https://www.jstor.org/stable/3679439
Accessed: 15-05-2019 07:44 UTC

Paul Berg,
Robert Rowe, and
David Theriault
Institute of Sonology
Plompetorengracht 14–16
Utrecht, The Netherlands

SSP and Sound Description

Introduction

SSP (Sound Synthesis Program) is a computer sound synthesis program designed by G. M. Koenig. The program produces sound in real-time (i.e., after a relatively short calculation time). SSP opens an experimental field of sound description because it refers only to amplitude and time values. This biparametric approach allows a complete reexamination of the issue of sound description, in particular the relation of user input to sound output. The major part of this paper is devoted to this problem. The unusual approach of SSP, however, is not without foundation; its background and roots will be discussed briefly before our treatment of sound description. The meaning of sound description here is twofold: it describes (1) a sound that is to be made (specification) and (2) a sound that has been made (analysis). Synthesis techniques can be and are used as a basis for analysis and, conversely, analysis techniques can be and are used as a basis for synthesis.

A brief characterization of the approach in SSP is in order. First, predetermined compositional rules are used to build a sound with the basic units of amplitude values and time values. The rules are either various aleatoric procedures or the direct enumeration of a sequence of values. The time and amplitude values are specified. A selection is made from this initial supply using the rules. A selection is then made from this last selection, and the result is called a *segment.* The segments are selected (ordered) and auditioned. In this way, a composer produces a single sound that may vary in length from a few milliseconds to several minutes. Composition and sound description are the definition of the *supply* (of amplitude and time values) and the series of selections made from it. A more extensive description of the program is found in the Appendix.

This approach to sound synthesis differs from the approach in many well-known computer sound synthesis programs. It does not refer to frequencies, spectral components, pulse-widths, modulation indices, unit generators, or any of a number of other key words. It has nothing to do with additive or subtractive synthesis. It would be difficult to produce familiar melodies or mood music (electronic wallpaper, as it were) with it.

SSP is not, however, intended as a reaction to present activities. Rather, it is an outgrowth of certain musical ideas—ideas that led to the stark realization that one could limit oneself to describing sound structures using amplitude and time values as units. Two notions are important in this regard:

1. The application of instrumental experience in macro-time (rhythmic relationships among parameter values) to micro-time (timbre formational laws)
2. The relationship between artistic means and ends. "Each art-form exploits its special production methods in order to endow the phenomena with unmistakable characteristics. Artistic economy demands that the means be appropriate to the end and the exploitation of the means be an end in itself." (Koenig 1971, p. 69)

The roots of these ideas can be found in the serialism of the 1950s and in the concept of *programmed music* (Koenig 1978).

Background

The work of the European serial composers in the 1950s is well known and was documented in their journal, *die Reihe.* The group clustered in Cologne. There they tried to produce electronic music unifying the macrostructure of the composition and the microstructure of the sound. An example of this approach is Koenig's *Essay* (1960). The sounds are not the realization of a preconceived acoustic ideal, but rather result from serial manipulation of basic material and subsequent ordered transformations.

Some conceptual contributions to SSP were made by serial music and its composers. Two of these concepts are

1. Composing sound using musical procedures instead of trying to reconstruct a sound based on analytical data
2. Defining compositional fields that change in time (tendency masks)

Both of these concepts are used in SSP.

Another root of SSP is in programmed music. This is not just music resulting from a computer program. It can be considered the result of a rigorous adherence to compositional rules. The rigorous application of rules, of course, lends itself well to computer applications. Koenig's attempt at generalizing some compositional rules led first to Project 1 for instrumental composition (Koenig 1970*a*) and then to Project 2 (Koenig 1970*b*).

In Project 2, the composer sets up a structure formula and has the computer work out one or more variants. The elements in Project 2 are the parameters: instrument, harmony, duration, entry delay, rest, register, dynamics, and performance mode. In Project 2, the concept of selections according to certain rules (selection principles) is used to determine the macrostructure of the composition. In SSP, the same selection principles are used to determine the microstructure of the sound.

The musical background of SSP can be seen, then, as European serialism (composing sound using procedures, tendency masks) and programmed music (using selection principles, selections of selections). Today, SSP better suits a composer who wants to define structures and listen to the results than a user who must have a certain sound. SSP leads one "to describe the composition as one single sound, the perception of which is represented as a function of amplitude distribution in time as sound and silence, soft and loud, high and low, rough and smooth" (Koenig 1978).

This brings us to the problem we want to address. How does one relate input and output with this approach when the user provides selection principles and values for them and the result is perceived as sound and silence, soft and loud, and so forth? Two approaches to this problem will be discussed.

1. The output can be described in terms of the input data that were used.
2. The output can be described using time and frequency domain signal processing techniques such as the Fast Fourier Transform.

Formation Analysis

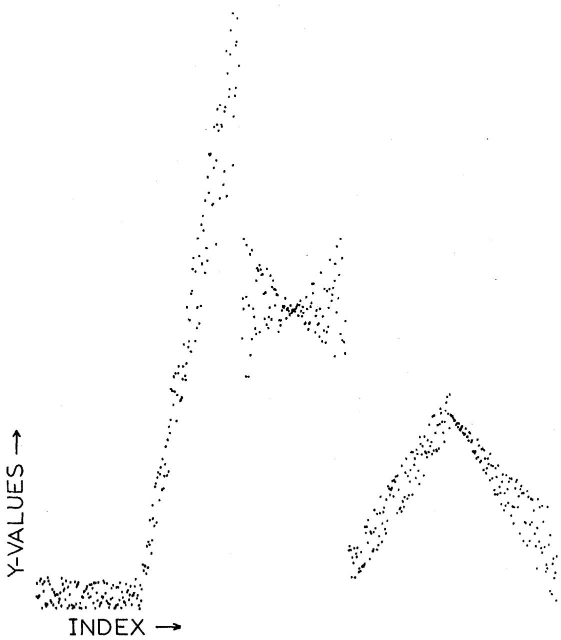
With the first approach, describing the sound output in terms of the input data, one could simply regurgitate all the input data and call that a description of the output. This is not the method discussed here. Instead, we distinguish five terms used to characterize the input data and, thereby, the resulting sound. These terms are

1. Expansion
2. Reduction
3. Reorder
4. Isolation
5. Copy

On each level of activity in SSP there is a supply of data; a selection principle is applied to this data, producing an output of new data. Each of the descriptive terms listed can be realized by a selection principle. Thus the terms specify how the selection principles may be used. For example, starting from one supply of data, the selection principle TENDENCY may be fed various input values that may cause it to exhibit the behavior described by each of the terms. Figure 1*a* shows the data constituting the supply, while Figs. 1*b*–1*f* show how each of the terms may be realized by the selection principle TENDENCY, and what the resulting output looks like graphically. The case given here is the most simple one; only one level of the hierarchy of SSP is used, and one parameter and one term at a time are treated. For the most part, any selection principle (ALEA, SERIES, RATIO, TENDENCY, GROUP, COPY) may "perform" any of the terms (expansion, reduction, reorder, isolation, copy). As our example showed, TENDENCY could copy (Fig. 1*b*), expand (Fig. 1*c*), reduce (Fig. 1*d*), reorder (Fig. 1*e*), and isolate (Fig. 1*f*). Each selection principle lends its own distinctive "flavor" to the result. The descriptive nature of the terms is not weakened in any case.

Fig. 1. Simple transformations of data structures complying to the five descriptive terms. Y may refer either to amplitude or to time values.

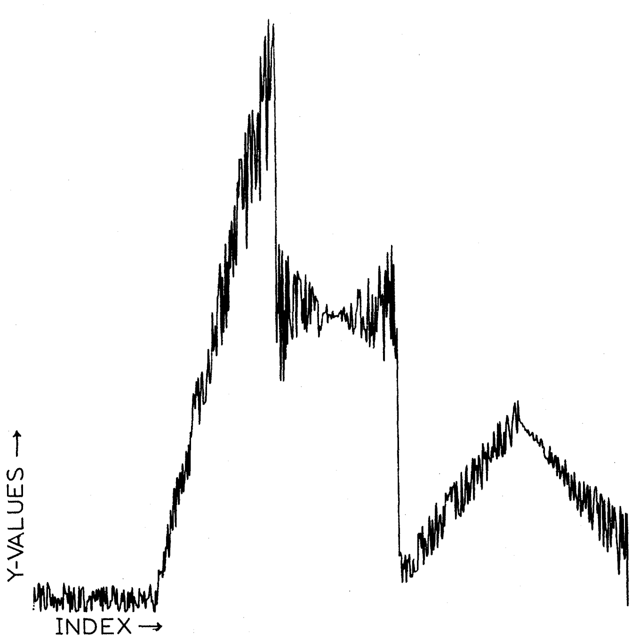
1a



1a Supply

TEND
N,M: 600,5
NN,A1,A2,Z1,Z2: 120,1,100,1,100
NN,A1,A2,Z1,Z2: 120,1,100,1500,2000
NN,A1,A2,Z1,Z2: 120,700,1200,1200,700
NN,A1,A2,Z1,Z2: 120,100,250,450,600
NN,A1,A2,Z1,Z2: 120,550,551,1,300

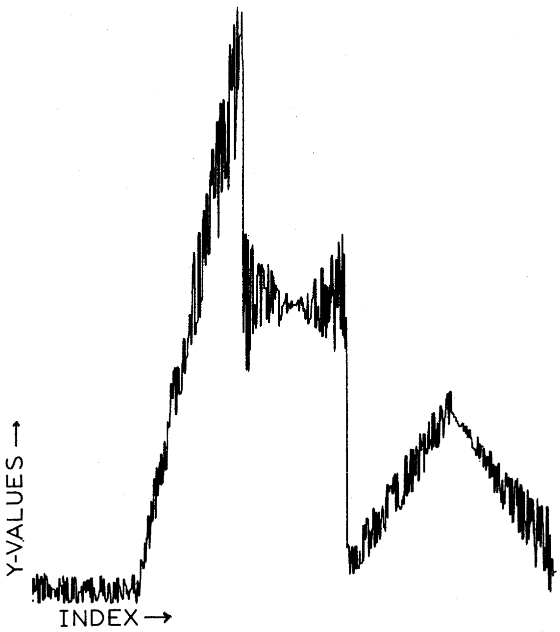
1b



1b Copy

TEND
N,M: 600,1
NN,A1,A2,Z1,Z2: 600,1,1,600,600

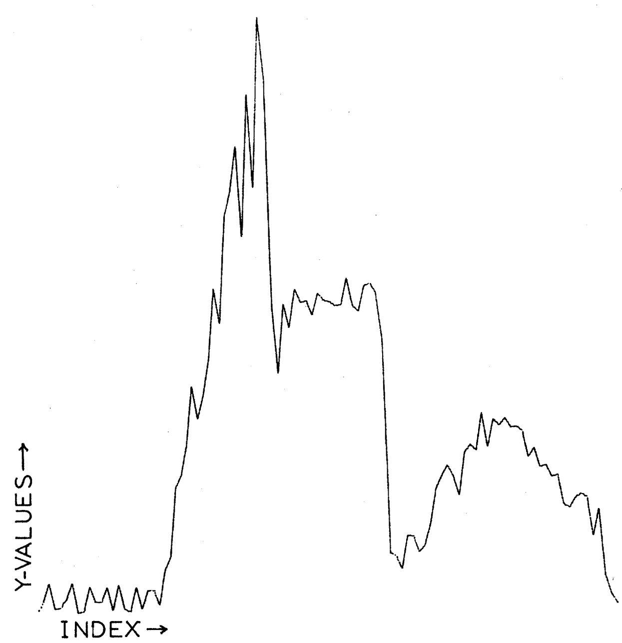
1c



1c Expansion

TEND
N,M: 1200,1
NN,A1,A2,Z1,Z2: 1200,1,2,599,600

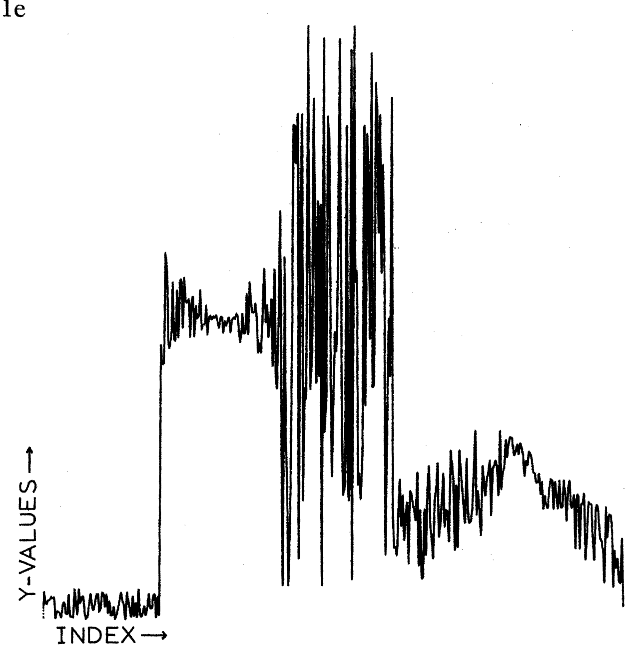
1d



1d Reduction

TEND
N,M: 100,1
NN,A1,A2,Z1,Z2: 100,1,2,599,600

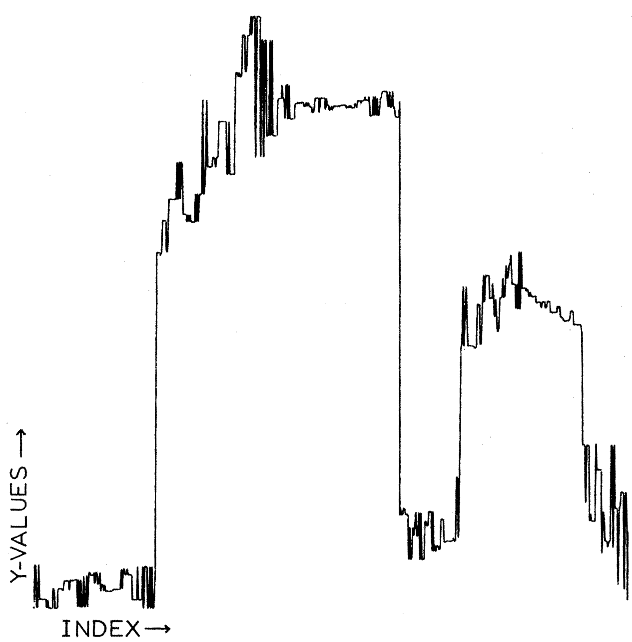
1e



1e Reorder

TEND
N,M: 400,5
NN,A1,A2,Z1,Z2: 80,1,120,1,120
NN,A1,A2,Z1,Z2: 80,241,260,340,360
NN,A1,A2,Z1,Z2: 80,121,240,120,240
NN,A1,A2,Z1,Z2: 80,361,420,420,480
NN,A1,A2,Z1,Z2: 80,481,500,580,600

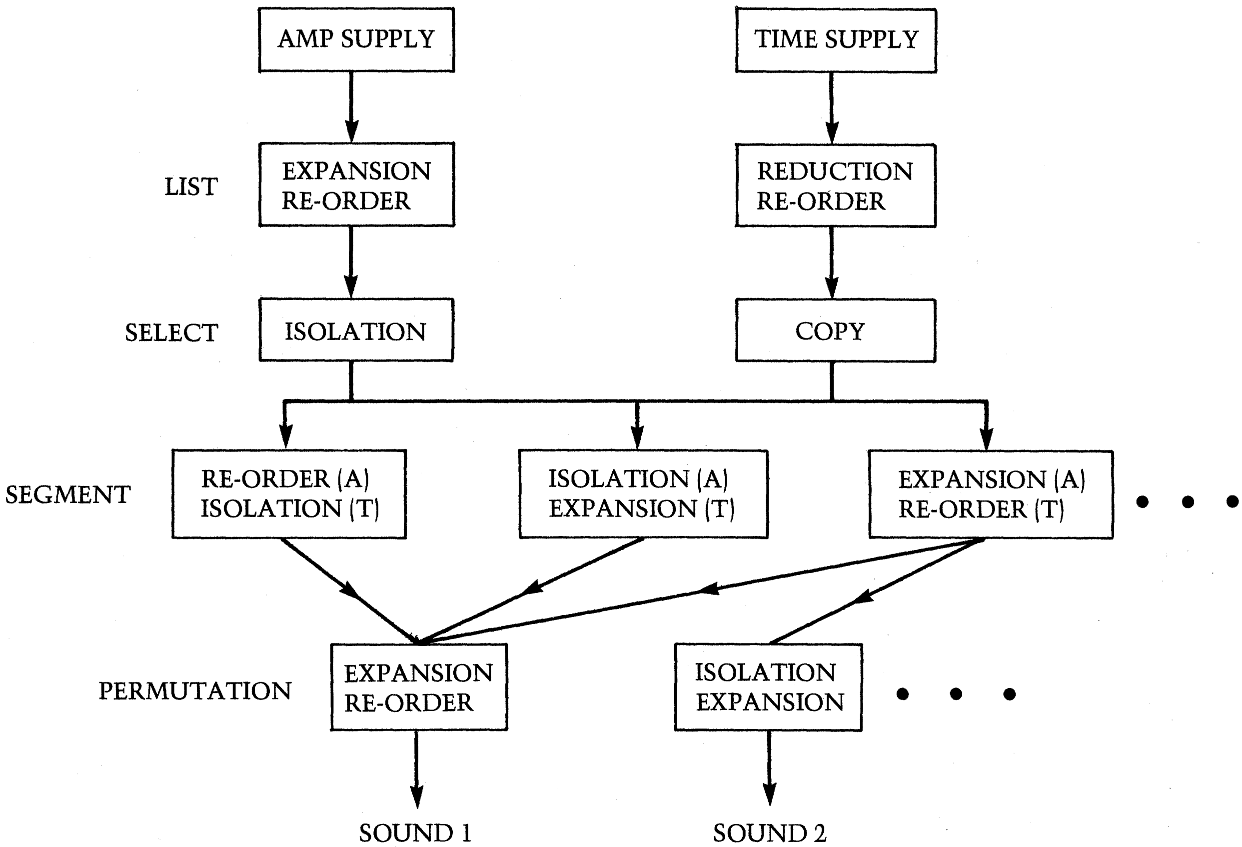
1f



1f Isolation

TEND
N,M: 600,7
NN,A1,A2,Z1,Z2: 120,50,51,70,71
NN,A1,A2,Z1,Z2: 120,170,171,190,191
NN,A1,A2,Z1,Z2: 120,290,291,310,311
NN,A1,A2,Z1,Z2: 60,361,362,380,381
NN,A1,A2,Z1,Z2: 60,460,461,479,480
NN,A1,A2,Z1,Z2: 60,481,482,500,507
NN,A1,A2,Z1,Z2: 60,580,581,599,600

Fig. 2. Formation plan and analysis.



We wish to emphasize that these descriptive terms are constructs; they form a conceptual basis whereby a user may interpret his or her actions. These terms are manifestations of the way the selection principles may be handled. That is, they refer directly to the transformation of data structures in SSP. They are not directly linked to the software.

Functional Hierarchies of SSP

In Fig. 2, one can see that the sound output is the result of the actions on the data supplies by selection principles (characterized in our descriptive terms here) on four functional levels:

1. LIST—an amplitude list and a time list are defined; these lists are source material for every other level of processing
2. SELECT—a selection is made from the elements specified in LIST
3. SEGMENT—a segment is an equal number of amplitude values and time values chosen from those selected in SELECT
4. PERMUTATION—the performing order of the segments is established

(There is actually an implicit fifth level, enabling one to play a given collection of segments ordered by the PERMUTATION level through a D/A converter and loudspeakers; one can also obtain numerical printouts or graphic plots at any level of the functional hierarchy.)

The chart in Fig. 2 is both an example of a working method and a description of the sound output. The working method can be called the *formation plan.* The description of the sound output can be called the *formation analysis.* The terms in the boxes are the actions to be performed. With a formation plan, the user structures the input data. Clearly, this could be exploited compositionally. This idea has a great deal of potential, and is only mentioned here to point out the possibility. What we want to focus on is formation analysis.

Formation Analysis Example

In Fig. 2, the LIST level expands and reorders the amplitude supply. On the SELECT level, one part is selected by isolation. (This is, of course, not the same thing as if LIST had contained expansion, reorder, and isolation.) Isolation in SELECT is performed on the result of the LIST activities; thus a hierarchical structure is formed. On the SEGMENT level there is again an opportunity to manipulate data, but now AMP and TIME are synchronized. A segment can then be categorized according to the actions the user has performed on three levels (LIST, SELECT, SEGMENT) in two parameters (AMP, TIME). The SEGMENT level can be extended horizontally; when many segments are made, relationships and comparisons can be drawn in such a way that a structured direction can be given. By using the constructs while designing the segment data configurations (formation plan), the user is able to structure the supply for the PERMUTATION level.

The final (PERMUTATION) level activities can be interpreted in the same manner; the supply this time is individual sound units (segments). The output result can thus be described as a data configuration/sound event designed in accordance with user constructs (expansion, reduction, etc.) from structural supplies in hierarchical relationships. The terms are used in a formation plan; the formation analysis uses this as a reference point.

The main advantage of formation analysis lies in a comparative usage. By listening to an individual PERMUTATION output and defining the structure as the sum of actions taken on various levels, the user can locate the output on a multidimensional grid of all possible results. For example, one may have specified data structures using the constructs for formation and then judged the sound output by the same terms. One then wishes to make some subsequent, related structures as a different permutation. One may use a different descriptive term to characterize data on any of the three levels in either parameter or in the PERMUTATION level. The analysis terms allow the user to easily follow the unfolding structural development from a comparative standpoint. Each resultant sound structure has its reference point in the previous one.

Sound Analysis

The sound output can be described in a second way by using signal processing techniques such as the Fast Fourier Transform. Two examples of how this can be done will be presented. In short, a number of related segments are produced and analyzed, and their analyses compared.

The first example (Fig. 3) deals with the GROUP selection principle as it was used on the SEGMENT functional level. GROUP is a useful tool for the creation of a wide palette of sounds; fine gradation can be made through the adjustment of four variables. These are (1) the interval from which values are to be chosen, (2) the total number of values desired, (3) the "type" of selection, and (4) the interval from which group sizes may be chosen. In this example the last variable, the group size interval, was chosen.

An amplitude list of the following 12 values was specified:

0 500 1000 1500 2000 2500 3000
3500 4000 3000 2000 1000.

Five segments were made using GROUP in the amplitude parameter; time values were kept constant. The first five data items of the GROUP specifications were also constant, that is, A, Z, N, TYPE,LA: 1,12,55,4,1. The upper limit of the group size interval was changed per segment, including the values 2,4,6,8, and 10. N and the maximum group size interval were chosen such that the last segment would exhaust the pool of possible group sizes.

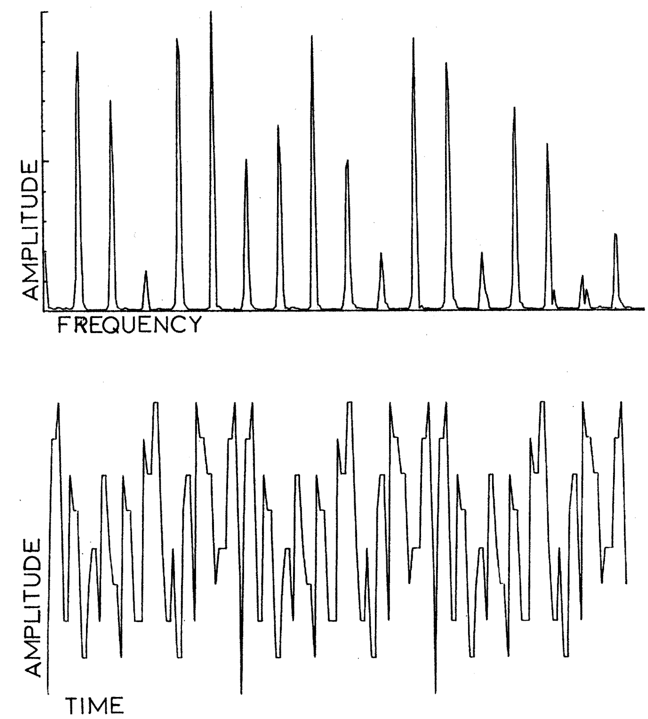
Examination of the Fourier analyses of these sounds showed that an increased group size interval

Fig. 3. Fourier Transform and signal function of segments constructed with GROUP.

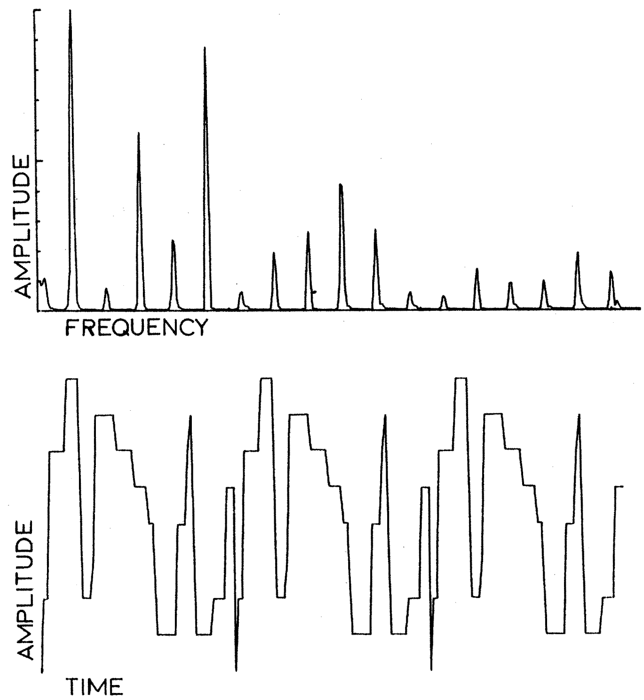
SEGMENT 1

SEGMENT 3

SEGMENT 1 (Fourier Transform, signal function)
GROUP
A,Z,N,TYPE,LA,LZ: 1,12,55,4,1,2



SEGMENT 3 (Fourier Transform, signal function)
GROUP
A,Z,N,TYPE,LA,LZ: 1,12,55,4,1,6



resulted in the concentration of the sound's energy in the lower partials. The signal was increasingly focused on one partial as the series progressed, though the partial which was predominate changed through the last segments. All five segments, when played with multiple periods on the SOUND functional level, exhibit the same fundamental frequency of 344 Hz, which stems from the constant segment length.

It is apparent from plots of the signal functions that this concentration of energy results because the larger group sizes allow fewer up-and-down excursions of the waveforms. When the group size interval includes larger values, the exact number of excursions is unpredictable because it depends on the value selected: the repetition of a value adjacent to the last one chosen will not produce a major excursion (see Fig. 3, SEGMENT 3). When the group size interval summation is larger than N, a random deletion of values will occur because some groups will be curtailed or eliminated at the end of a segment. Because of this, the concentration of energy that occurs in this test continues less predictably in examples involving oversaturation. In this case, the random deletion of values precludes certainty about the nature of waveform excursions.

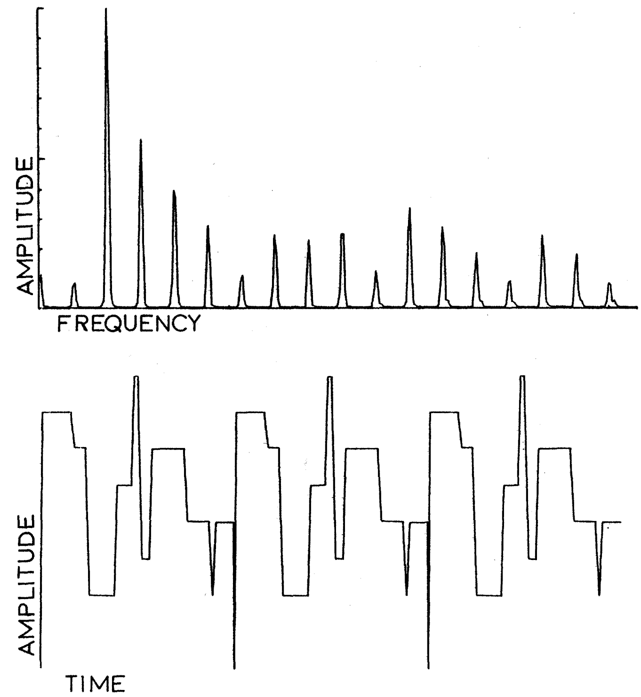
The second example describing the output using Fourier analysis involves the selection principle RATIO. In RATIO, the selection can be weighted in favor of a limited number of elements. Given the following amplitude values:

4 341 1448 1930 3175 3532 3879
3118 1790 1000 467 7

12 segments were formed using RATIO in such a way that one amplitude point occurred nine times

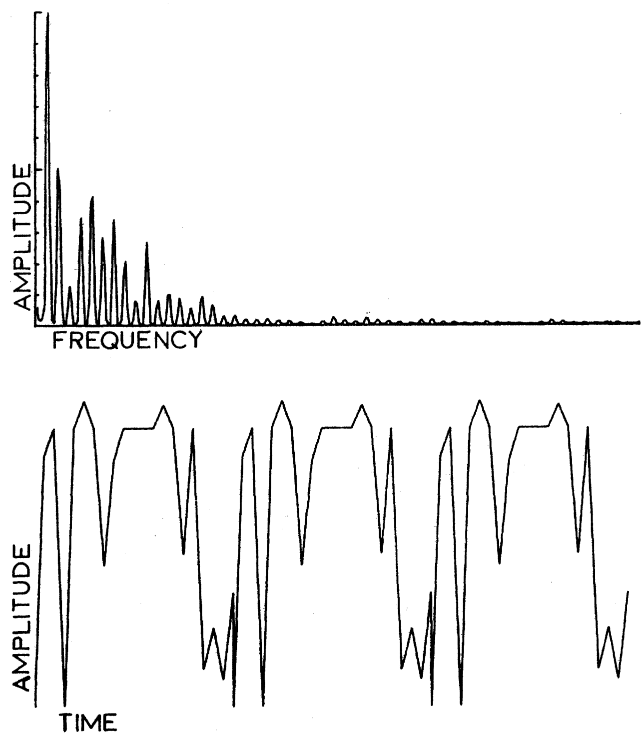
SEGMENT 5

SEGMENT 5 (Fourier Transform, signal function)
GROUP
A,Z,N,TYPE,LA,LZ: 1,12,55,4,1,10



SEGMENT 6

SEGMENT 6 (Fourier Transform, signal function)
RATIO
FACTORS: 1,1,1,1,1,9,1,1,1,1,1,1
A,Z,N: 1,12,20



more often than any other. To emphasize the first amplitude point (4), the data was:

FACTORS: 9,1,1,1,1,1,1,1,1,1,1,1
A,Z,N: 1,12,20

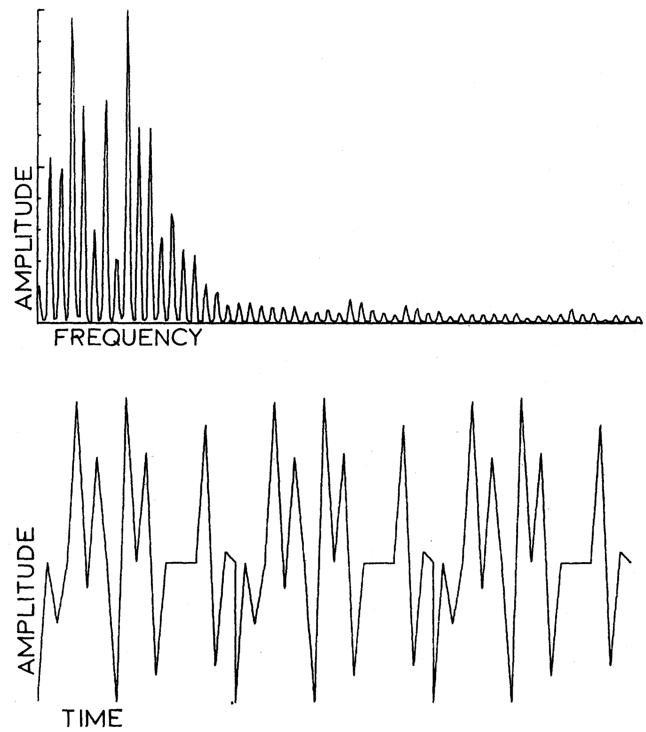
Each of the succeeding segments was made in the same way. The emphasized amplitude points correspond in their SELECT index to the segment number, that is, segment 1 emphasizes SELECT index 1 (4), segment 2 emphasizes SELECT index 2 (341), and so forth. This procedure yielded 12 segments that had the same fundamental frequency: 111 Hz, ± 2 Hz. This stems from common total duration of segments with constant time values.

The sound quality of the segments produced with one emphasized amplitude had more to do with the deviations from this defined norm than with the norm itself. Here, *deviations* refer to those amplitude values other than the one emphasized. The sounds have basically the same fundamental and are all harmonic, but the distribution of energy in the spectrum and the strongest formant exhibit radical change from segment to segment. Examination of the signal functions and Fourier analyses show that the complexity of the sounds has to do with the frequency of deviations. That is, if a segment maintains the main amplitude repetitively and groups the deviations in a similar manner, the signal splits into a few recognizable parts and thus creates a strong formant at a low partial (Fig. 4*a*, segment 6). If a segment alternates the primary amplitude and its deviants more rapidly, the greater number of excursions produces a more complex signal (Fig. 4*b*, segment 10).

Fig. 4. Fourier Transform and signal function of segments constructed with RATIO.

SEGMENT 10

SEGMENT 10 (Fourier Transform, signal function)
RATIO
FACTORS: 1,1,1,1,1,1,1,1,1,9,1,1
A,Z,N: 1,12,20



Summary

SSP provides the composer with an interesting, experimental framework for manipulating sound structures. The sounds produced result from the invocation of a collection of compositional rules on four levels. The specific behavior produced by these rules may be organized prior to their invocation by means of a technique of description called a formation plan. The resulting sound may also be described in these terms; such a description is called a formation analysis. Alternatively, one may use sound descriptions derived from acoustics and signal processing, such as a characterization in the frequency domain by means of the Fast Fourier Transform.

Acknowledgments

This paper is based on portions of *Sonological Reports No. 5,* "SSP—A Bi-parametric Approach to Sound Synthesis" (132 pages) by J. D. Banks, P. Berg, R. Rowe, and D. Theriault. The graphs of the spectral analyses were produced with a Fast Fourier Transform program written by S. Tempelaars. The authors wish to acknowledge the aid of *Computer Music Journal* editor C. Roads in preparing this manuscript for publication herein.

References

1. Banks, J. D.; Berg, P.; Rowe, R.; and Theriault, D. 1979. SSP–A bi-parametric approach to sound synthesis. *Sonological reports no. 5.* Utrecht, The Netherlands: Institute of Sonology.
2. Berg, P. 1978. *A user's manual for SSP.* Utrecht, The Netherlands: Institute of Sonology.
3. Koenig, G. M. 1960. *Essay, composition for electronic sounds.* Vienna: Universal Edition.
4. Koenig, G. M. 1970*a*. Project 1. *Electronic Music Reports* 2:32–44.
5. Koenig, G. M. 1970*b*. Project 2, a programme for musical compositon. *Electronic Music Reports* 3.
6. Koenig, G. M. 1971. *Summary: observations on compositional theory.* Utrecht, The Netherlands: Institute of Sonology.
7. Koenig, G. M. 1978. Composition Processes. Lecture delivered at the UNESCO Workshop on Computer Music, Aarhus, Denmark.

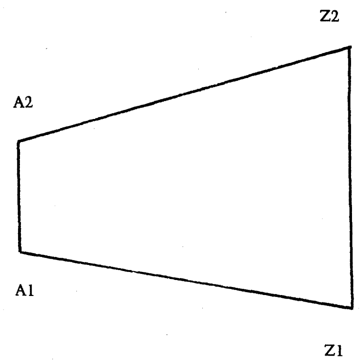
Appendix—A Brief Description of the Program

The User's Task

An informal description of the user's task follows:

1. List the source material (amplitude values, time values).
2. Select all or part of that source material and put it into a "working area."
3. Construct segments from the material in the working area. Segments contain an equal number of amplitude and time values.
4. Order the segments.

Fig. 5



5. Listen to the chosen segments in the chosen order.
6. Return to 1 or 2 or 3 or 4.

These actions are called *functions,* which take place on functional levels (e.g., the SELECT level). The rules for carrying out the above functions are *selection principles.* Selection principles are applied to *elements.* Elements are amplitude or time values (in function LIST), indices of amplitude and time values (in functions SELECT and SEGMENT), and segment numbers (in function PERMUTATION).

The program is conversational: questions are printed on the terminal and the user types responses. A protocol of all user responses is recorded in a protocol file for later reference.

Selection Principles

The available selection principles are ALEA, SERIES, RATIO, TENDENCY, GROUP, and COPY. In the following definitions all values are integers.

ALEA—N random values are chosen between A and Z.
SERIES—N random values are chosen between A and Z. When a value has been selected, it is removed from the pool of available values from which the selection is made. When the pool contains no more values, it is refilled with all the values between A and Z.
RATIO—N random values are chosen between A and Z. The ratio of the likelihood of occurrence of values between A and Z is specified as factors.
TENDENCY—N random values are chosen between boundaries that change in time. M number of masks may be specified. NN values are chosen in one mask that is indicated by initial boundaries (A1,A2) and final boundaries (Z1,Z2) (Fig. 5).
SEQUENCE—A sequence of numbers is given in order.
COPY—The available values are copied.
GROUP—A random value between A and Z is chosen. This number is repeated one or more times in succession to form a group. The size of the group is a random value between LA and LZ. The value between A and Z as well as the group size between LA and LZ may be chosen with either ALEA or SERIES. The combination of ALEA and SERIES is indicated with TYPE. In total N values are selected.

Functions and Functional Levels

Functions perform certain tasks. In most cases, selection principles are chosen to perform these tasks.

LIST—A list of amplitude values and a list of time values are defined. Amplitude values may be between 0 and 4095 (presently, a 12-bit converter is used). Time values are expressed in microseconds (minimal value 38 μs).
SELECT—A selection is made from the elements chosen in LIST.
SEGMENT—One or more segments are defined. A segment is an equal number of amplitude and time values chosen from those selected in SELECT.
PERMUTATION—The order of the segments is established.
SOUND—The segments are "performed" as ordered in PERMUTATION. This period may be repeated the desired number of times.
PRINT—The numerical results of various functions can be printed.
PLOT—The results of certain functions can be plotted.
STATUS—The number of elements in each file created by a function is listed.
HALT—The program stops.